**《人工智能基础A》大作业报告**

**Pycharm可视化开发综合应用实践**

一、实验目的

通过实验，学会Pycharm可视化开发。具体目标要求如下：

1） 学习PyQt5图形界面开发

2） 掌握Qt Designer与pyuic工具

3） 综合运用PyQt5和深度学习技术，开发一个人脸情感图像识别模型。

二、实验过程

1. pycharm下配置pyqt5

1.1 创建conda虚拟环境

为了避免依赖冲突和管理不同的环境，首先创建一个新的虚拟环境：

cond create -n env\_name python=3.121.2 安装PyQt5

install -i https://pypi.tuna.tsinghua.edu.cn/simple/ pyqt5

pip install -i https://pypi.tuna.tsinghua.edu.cn/simple/ pyqt5-tool

1.3在pycharm中配置外部工具

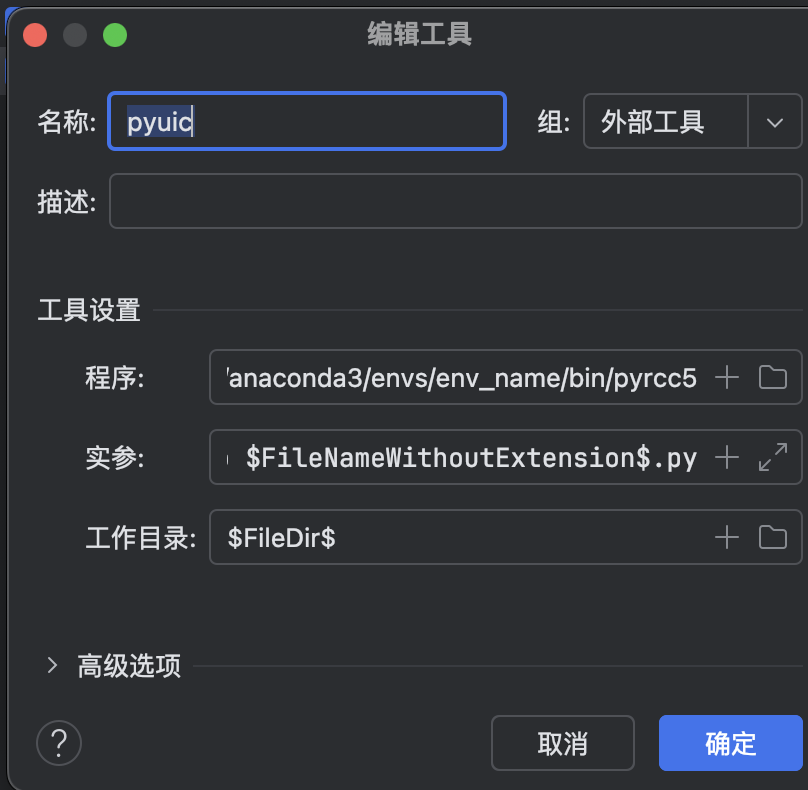
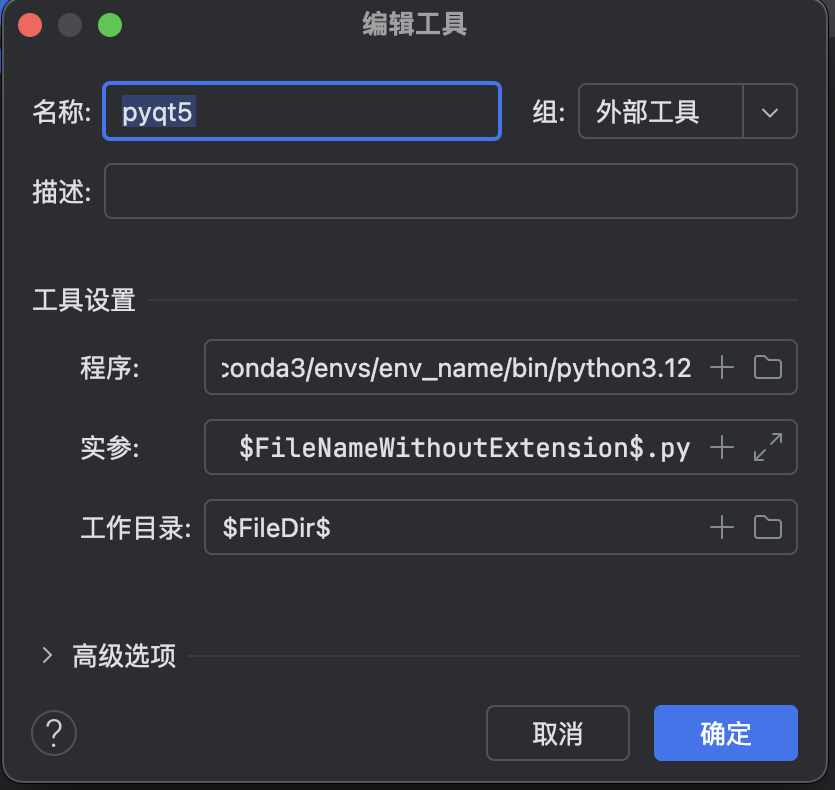
打开 PyCharm，并选择第一步中创建的虚拟环境作为默认 Python 解释器。接下来，打开 PyCharm 设置，进入 “外部工具” 配置页面，点击“+”添加新的外部工具：

工具名称：pyuic5

工具路径：<your\_conda\_env>/Scripts/pyuic5.exe（根据实际路径调整）

参数：-o $FileNameWithoutExtension$.py $FilePathRelativeToProjectRoot$

这样可以方便地将 .ui 文件转换为 .py 文件。

2.ui设计

使用 Qt Designer 设计用户界面，包括窗体、按钮和其他控件：

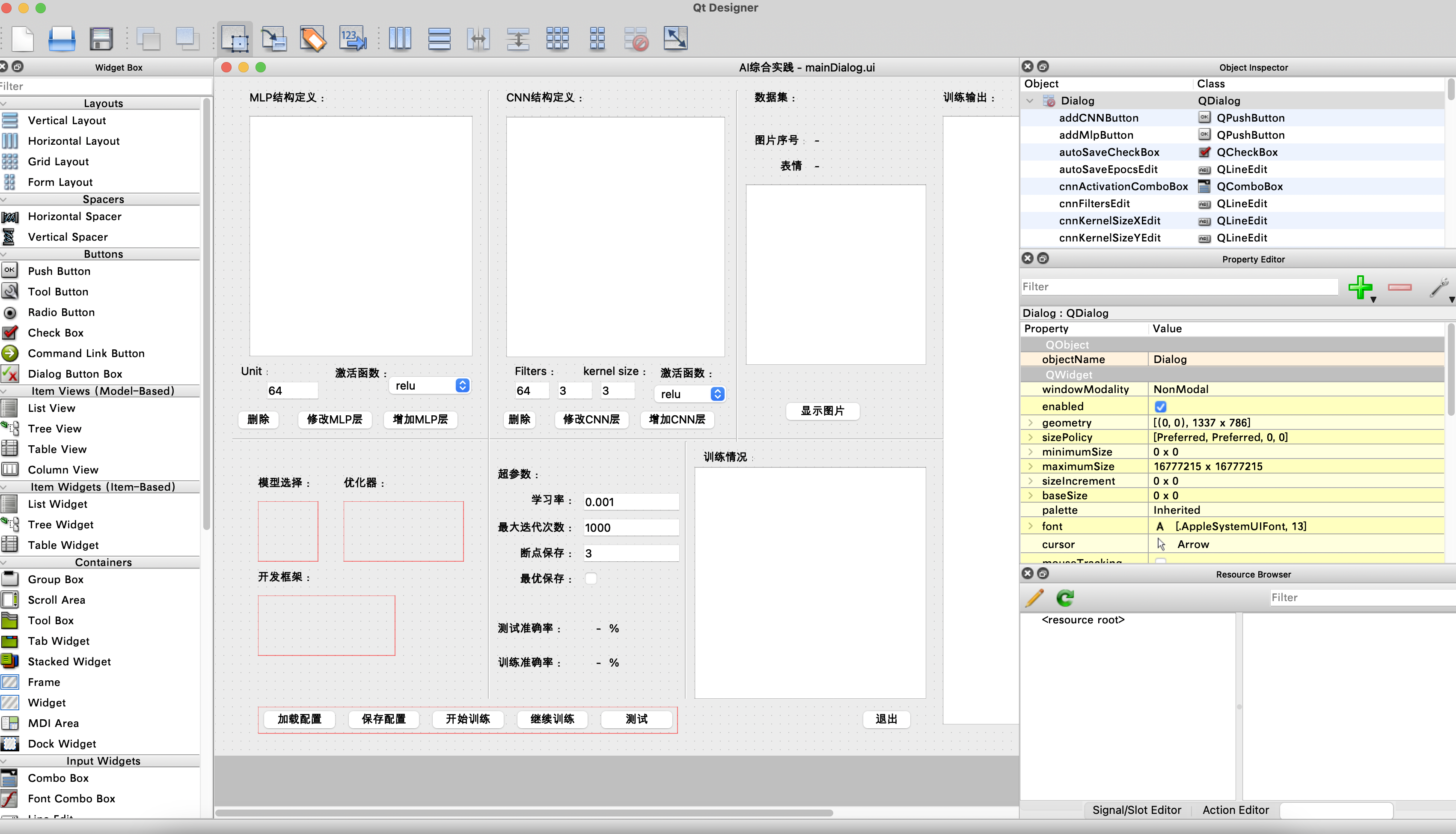
打开 Qt Designer 工具，创建一个新的窗体。

设计界面布局，放置按钮、文本框、图像显示区等控件。

确定界面的交互逻辑，例如：

按钮点击事件触发数据加载。

图像展示区域用于显示人脸图像。

文本框或标签用于显示模型训练的结果（如准确率、损失值等）。

设计完成后，将.ui文件保存。

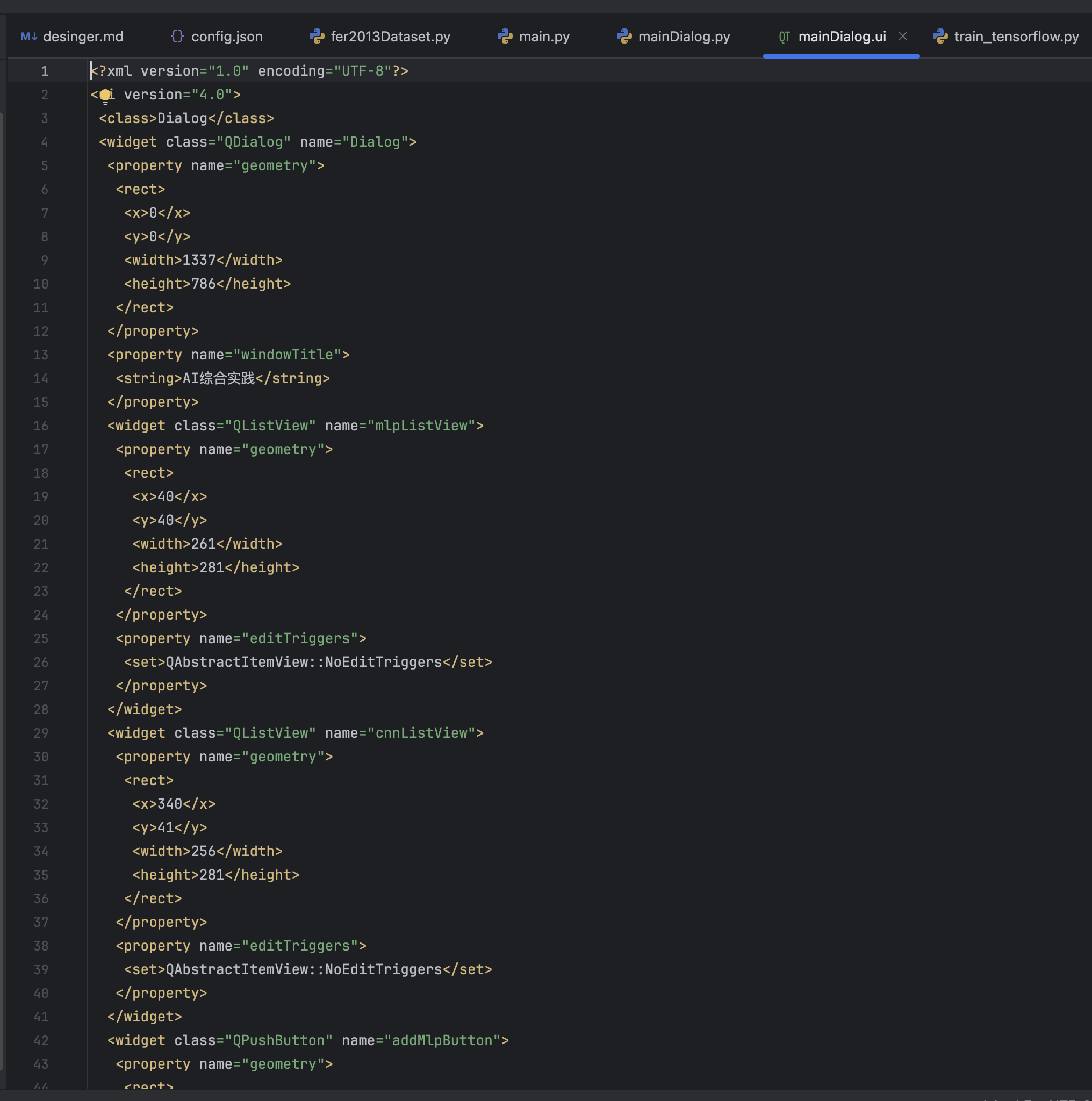
3. 使用 External Tools – PyUIC 转换 .ui 文件为 .py 文件

在 PyCharm 中，使用 pyuic5 工具将设计好的 .ui 文件转换为 Python 代码

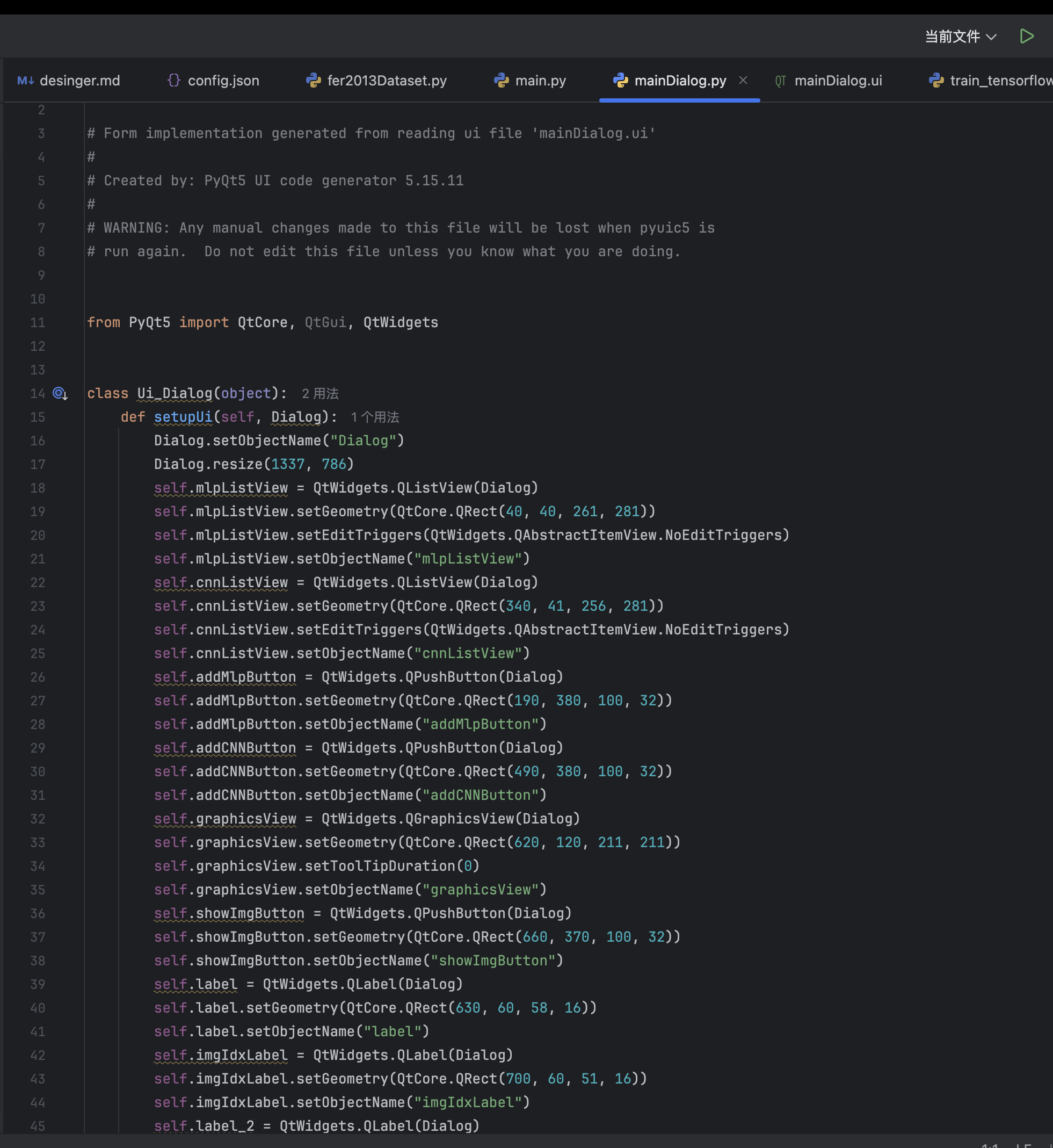
在 PyCharm 中，右击 .ui 文件，选择 “External Tools -> pyuic5” 进行转换。

转换后会生成一个 .py 文件，该文件包含了界面设计的 Python 实现代码。

以下为ui文件：

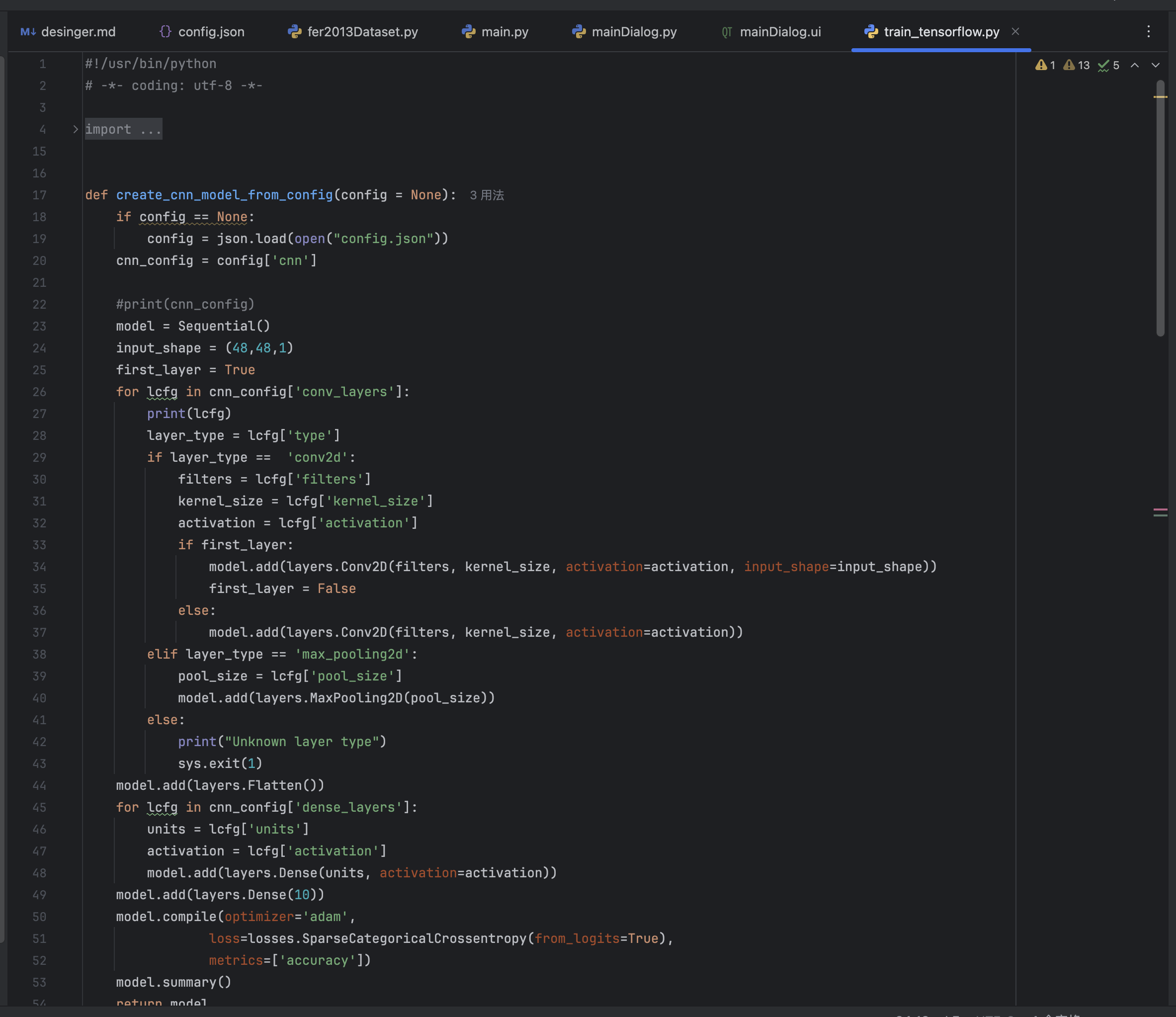


以下为py文件。



4.设计CNN模型，Tensorflow框架下的机器学习模型

使用 TensorFlow 设计一个基本的卷积神经网络（CNN），包括多个卷积层、池化层和全连接层：



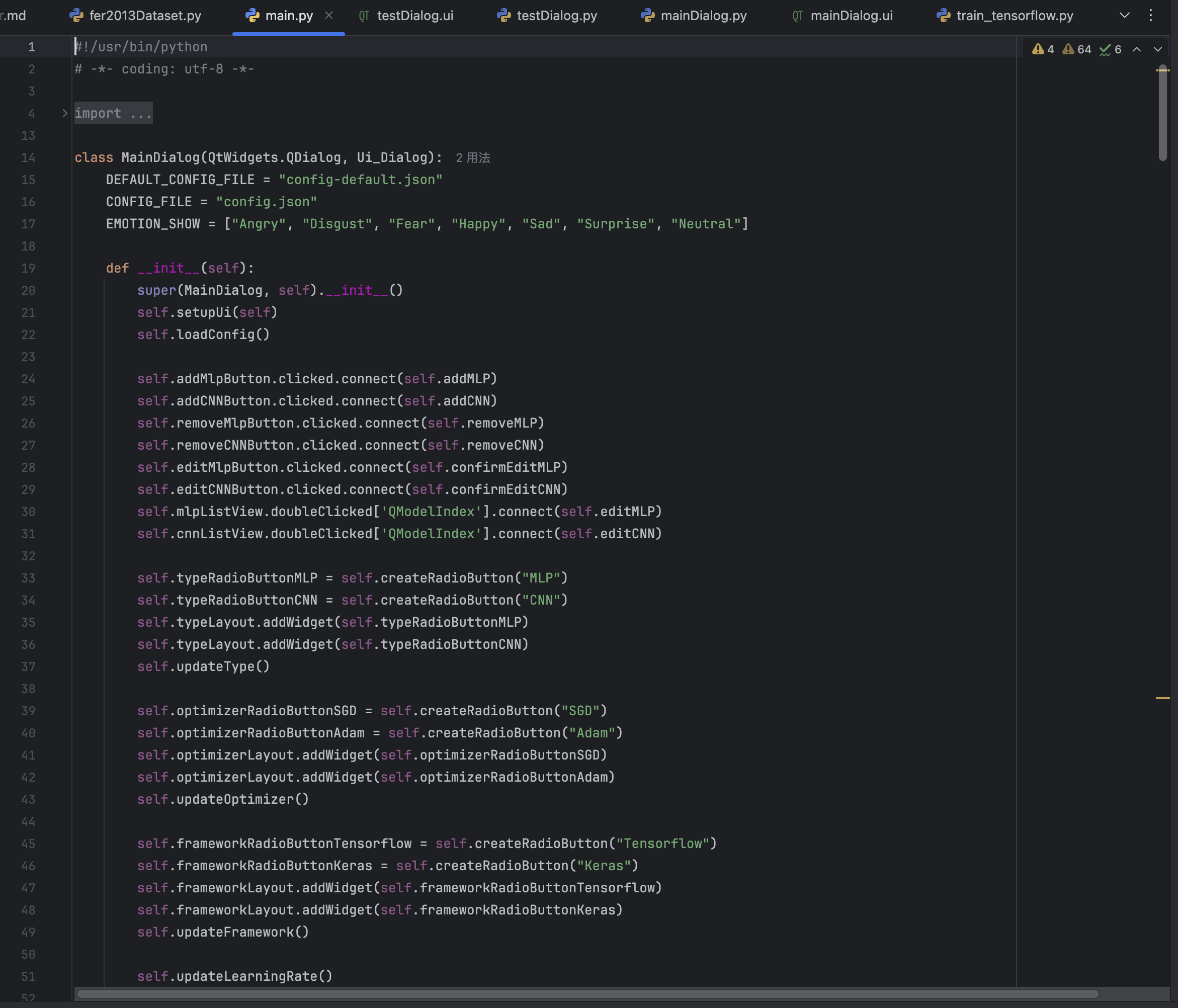
4.运行主程序

在 PyQt5 界面中，通过按钮点击来加载数据、开始训练、显示结果。

在数据加载阶段，用户可以浏览并选择数据集。

在训练阶段，展示训练的实时进度，包括损失值、准确率等。

在训练结束后，展示最终的训练结果，并通过图表（如 ROC 曲线）展示模型的性能。



加载数据的过程如图



可以在数据集窗口浏览用到的数据集：



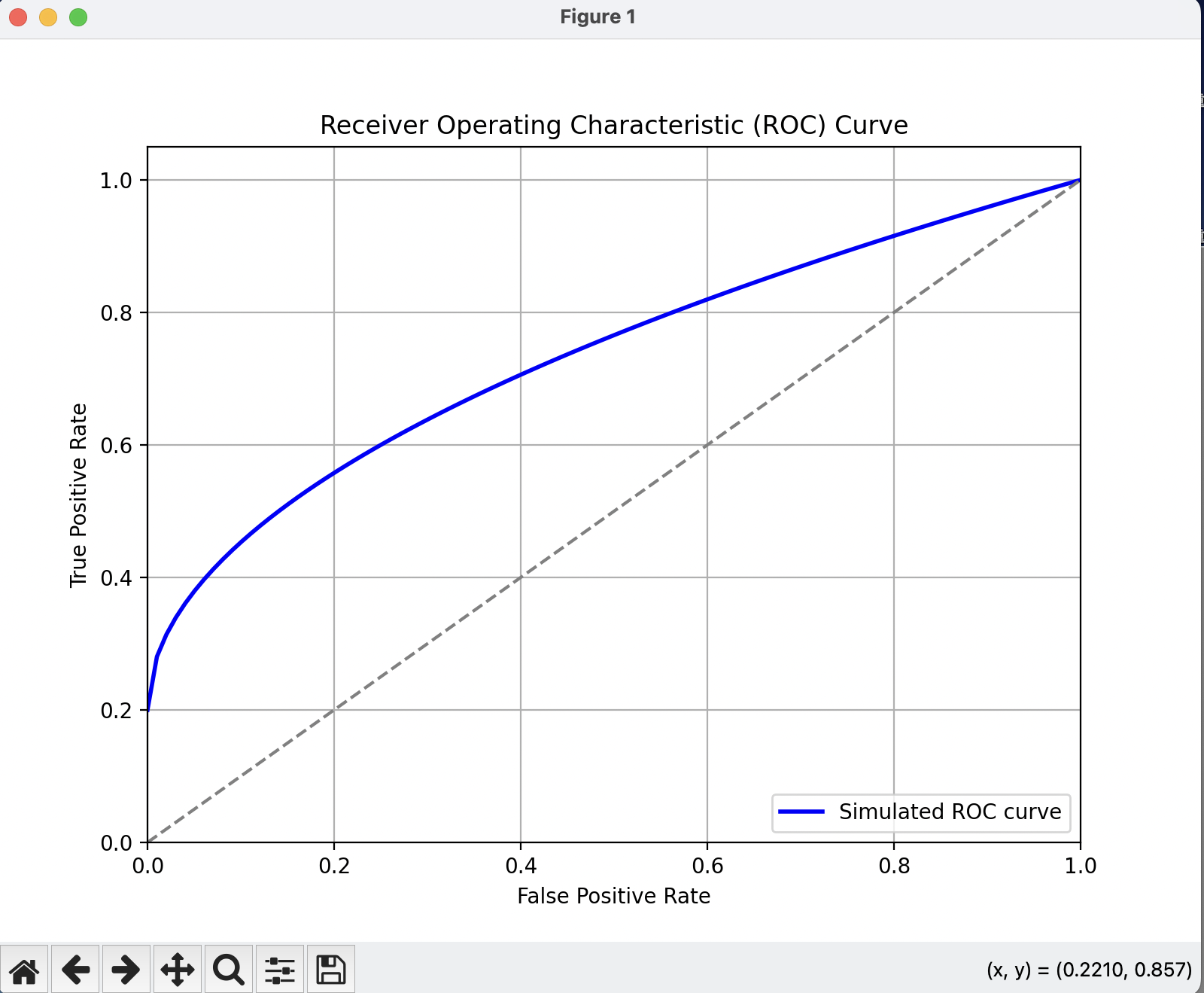
加载数据完成，开始训练，以下是训练过程：



训练和测试结果：



绘制ROC曲线：



三、实验总结

1．PyQt5 可视化开发：通过本次实验，我深入学习了如何使用 PyQt5 设计和开发图形界面，并掌握了如何使用 Qt Designer 和 pyuic5 工具将 .ui 文件转化为 Python 代码。这些技能将有助于我在未来开发更多的桌面应用程序。

2. 深度学习技术应用：在深度学习部分，利用 TensorFlow 实现了一个卷积神经网络（CNN）模型，并用它对人脸情感数据集进行了训练。通过该实验，我更加熟悉了模型的设计、训练及评估过程。

3. ROC 曲线的意义：ROC 曲线帮助我直观地理解了模型在不同阈值下的表现，